REMARKS

I. Comments on the Restriction Requirement posed in recent action

In the action mailed February 26, 2008, the Examiner withdrew new claims 107-109 as allegedly being drawn to a non-elected invention and has thus has required restriction between product and process claims. Applicants respectfully request that should a product claim subsequently be found allowable, withdrawn process claims that depend from (or otherwise include all the limitations of) the allowable product claim be rejoined in accordance with the provisions of M.P.E.P. § 821.04.

II. The Rejection under 35 U.S.C. § 112, first paragraph, should be withdrawn.

In the action, the Examiner maintained the rejection of claims 88, 89 and 91-96 and 101-106 as allegedly failing to be supported by an enabling disclosure. Applicants request reconsideration of the rejection in view of the following remarks.

A. Claims 88, 89 and 91-96

The Examiner acknowledges enablement of antibodies that bind to proteins encoded by SEQ ID NO: 1, 5, 9, 11, 13 and 15. See page 4 of the action. It is not disputed that techniques of mutagenesis, recombinant expression and making antibodies are routine in the art. See page 6 of the action. Instead, the Examiner's rejection of the antibody claims appears to be based on the specification's alleged lack of enablement for the production of protein variants that retain the ability to decrease bone mineral content. The Examiner's position is that, without knowing where and what mutations can be made so that the protein retains activity, it would require undue experimentation to make a protein variant that exhibits the activity recited in the claims. See page 6 of the action.

For the reasons explained in further detail below, the specification does provide guidance regarding how to produce protein variants that retain the cysteine backbone and the ability to decrease bone mineral content, as recited in the claims. Given this guidance, it would require only routine experimentation to make such protein variants and produce antibodies that bind to such protein variants.

"The test [for undue experimentation] is not merely quantitative, since a considerable amount of experimentation is permissible, if it is merely routine, or if the specification in question provides a reasonable amount of guidance with respect to the direction in which the experimentation should proceed to enable the determination of how to practice a desired embodiment of the claimed invention." Johns Hopkins Univ. v. Cellpro, Inc., 152 F.3d 1342, 47 U.S.P.Q.2D 1705 (Fed. Cir. 1998). "The enablement requirement is met if the description enables any mode of making and using the invention." Johns Hopkins Univ., *supra*. (emphasis added).

Experimentation, even if extensive, is not necessarily undue if it is routine in the art. In re Wands, 858 F.2d 731 (Fed. Cir. 1988). In re Wands involved screening of large numbers of hybridomas to identify specific hybridomas that fell within the claim limitations. The court in Wands indicated that because Wands provided sufficient guidance to make and screen the hybridomas and presented working examples, the enablement requirement was fulfilled. In re Wands, 858 F.2d 731, 740 (Fed. Cir. 1988). In re Wands does not hold that a specific number of working examples is required. In reaching its decision, the court in Wands considered that the inventor's disclosure provides considerable direction and guidance on how to practice the invention and presents working examples. Id at 740. When such provided guidance is coupled with high level of skill in the art, the invention is enabled. Id.

The specification provides teaching that guides one of ordinary skill in the art where to make additions, substitutions or deletions, and how to screen the resulting variants for activity (e.g., the ability to decrease bone mineral content). For example, in the section describing how to make amino acid modifications to the protein, the specification states that the cysteine backbone of the protein (illustrated in Figure 1) should generally be conserved. See, e.g., page 21, line 7, page 26, line 29-page 27, line 1, and Figure 1.

Moreover, the specification provides seven different sequences of native mammalian (human, vervet, mouse, rat and bovine) and variant human cDNA encoding a protein that decreases bone mineral content (SEQ ID NOS: 1, 5, 7, 9, 11, 13 and 15, see pages 81-84 of specification), from which one can determine which amino acids and regions are conserved among mammalian species. Attached hereto as Appendix A is a sample alignment showing conserved residues among the coding regions of these mammalian sequences. The human

and rat sequences are about 87% identical; the human and murine sequences are about 88% identical, and the human and vervet sequences are about 97% identical. The sample alignment was generated using the current version of the CLUSTAL W (1.83) program, but earlier versions prior to the filing date (see, e.g., Higgins et al., "Using CLUSTAL for multiple sequence alignments," Methods Enzymol., 266, 383-402, 1996, set forth herein as Appendix B) could easily have been used to generate a similar alignment. Alternatively, visual inspection of the sequences would have yielded the same information.

From this information, one of ordinary skill in the art could easily make knowledgeable choices regarding modifications; for example, conservative substitutions in the conserved regions are more likely to retain activity, while non-conserved regions are better able to tolerate non-conservative modifications. Similarly, substitutions in the cysteine backbone that affect folding are more likely to reduce activity. Moreover, the specification at page 63, lines 18-27 discloses various methods to determine bone mineral content or bone density.

The discussion in the action fails to consider this guidance in the specification, i.e. retention of the cysteine backbone, and knowledge of conserved regions in other mammalian species and human variants (see the alignment in Appendix A). There is insufficient explanation or reasoning as to why this guidance in the specification does not provide a good "direction in which the experimentation should proceed," *Johns Hopkins Univ. v. Cellpro, Inc.*, *supra*, that enables one skilled in the art to produce a number of protein variants that retain the desired activity. There is also insufficient explanation or reasoning as to why it would be ***undue*** experimentation to produce such variants. The only explanation of the "undue" nature of the experimentation is the "large quantity" (see page 6 of action) of experimentation, which is not necessarily undue experimentation. See *In re Wands, supra.*

Moreover, the discussion in the action fails to consider the fact that Applicants have obtained an issued patent, with its associated presumption of validity, with claims that encompass protein variants of the scope recited in the claims. MPEP §1701 states that:

> Every patent is presumed to be valid. . . . Public policy demands that every employee of the United States Patent and Trademark Office (USPTO) refuse to express to any person any

> opinion as to the validity or invalidity of, or the patentability or unpatentability of any claim in any U.S. patent, except to the extent necessary to carry out [a reissue, reexamination or interference of that issued patent].

The Examiner has not provided specific reasons why a genus of polypeptides encoded by polynucleotides 90% identical to the exemplified sequences in the specification is not enabled. The scope of the polynucleotides recited in claim 89 parallels issued claims in grandparent application U.S. Patent No. 6,395,511. Claim 1 of this patent is set forth below:

> 1. An isolated nucleic acid molecule comprising a polynucleotide having at least 90% identity with the full length of SEQ ID NO:1 or the complement thereof, wherein said nucleic acid molecule encodes a protein which specifically binds to at least a human bone morphogenic protein selected from the group consisting of bone morphogenic protein 5 and bone morphogenic protein 6.

Moreover, Applicants bring to the Examiner's attention unrelated U.S. Patent No. 6,562,949, issued on May 13, 2003, in which claim 1 reads as follows:

> 1. An antibody that specifically binds polypeptide with an amino acid sequence that is at least 90% identical to the amino acid sequence of SEQ ID NO: 2, wherein the percent identity is calculated using the GAP program with a unary comparison matrix, a 3.0 gap penalty, an additional 0.10 penalty for each symbol in each gap, and no penalty for end gaps, and said polypeptide binds a semaphoring selected from the group consisting of A39 semaphorin and AAV semaphorin.

Applicants submit that this claim language is similar to the claim language under consideration in the present application. The percent identity quoted in the claim above relates to an amino acid sequence whereas the percent identity of Applicants' claims is expressed in terms of the encoding polynucleotide. Applicants cite this patent as proof that the Patent Office has previously taken the position that a *genus of antibodies* is enabled and adequately described by language that refers to a genus of polypeptides. Having adequately described and enabled a novel protein that is 90% identical to a reference sequence, it would require no more than routine experimentation to make antibodies to such proteins.

The Examiner further contends that the specific hybridization conditions recited in the claims allow a high degree of sequence variation. See page 7 of action. For the reasons discussed in Applicants' response filed August 21, 2007, in general if one assumes that a 1% mismatch of two DNAs lowers the Tm 1.4°C, then for SEQ ID NO: 1 it would be estimated that about 80% homology is required for successful hybridization under the wash conditions recited in claim 88. While the scope of the claims is not limited by any theoretical calculations such as this, this relatively high estimated homology contradicts the Examiner's assertion that the conditions recited in the claims "allow a high degree of sequence variation."

The Examiner also asserts that the specification provides little or no guidance as to which amino acids in the variant polypeptide can be changed while maintaining the antibody binding affinity recited in the claims. However, the Examiner's concern is misplaced because the general intent is not to mutate the protein to obtain a variant that binds to a specific antibody with the desired affinity. Instead, one skilled in the art would understand that the specification's disclosure is to use the protein variants in any of a variety of known techniques to generate antibodies to the protein(s), after which such antibodies can be screened for the recited binding affinity. As indicated in the specification at page 45, methods of determining binding affinity of an antigen to its antibody were well known in the art as of the filing date of the present application and thus screening for antibodies with the desired affinity is nothing more than routine experimentation for one of skill in the art.

Finally, Applicants note that the Examiner's concerns, with respect to the genus of protein variants to which the claimed antibodies bind, do not apply to claims 101-106, which either recite SEQ ID NO: 1 or polypeptides encoded by ***naturally occurring*** polynucleotides. It is not undue experimentation to use the known techniques of hybridization screening to screen any one of a number of mammalian or human DNA libraries for polynucleotides that encode orthologs or allelic variants of sclerostin, and a number of exemplary sequences obtainable in such a manner have been provided in the specification (SEQ ID NOS: 1, 5, 7, 9, 11, 13 and 15).

B. Claims 101-106

The Examiner objected to the recitation in claim 101 of binding to "a portion" of SEQ ID NO: 2 with a desired affinity. Applicants note that the original language of this claim describes the natural antibody-antigen interaction; in general antibodies of a desired affinity bind to a portion of, or an epitope of, a protein, rather than contacting and binding the entirety of a protein. Nevertheless, solely in order to expedite prosecution, Applicants have replaced this language with a recitation of binding to "a polypeptide encoded by SEQ ID NO: 1," an embodiment that the Examiner considers enabled according to page 4 of the action. This amendment of claim 101 to refer to the corresponding polynucleotide rather than the polypeptide does not change the scope of the claim but merely places it in the form suggested by the Examiner.

The Examiner also asserted that claim 102 encompassed polypeptides encoded by a non-coding strand of a polynucleotide. This rejection is moot in view of the amendments made herein. Specifically, claim 102 as amended herein recites that the polynucleotide is capable of hybridizing to the complement of SEQ ID NO: 1.

Finally, the Examiner contends that the specification fails to describe how to make and use antibodies that further comprise effector or reporter molecules. Contrary to the Examiner's assertion, the specification does indeed describe how to make antibodies comprising effector or reporter molecules. See, for example, pages 38-40 of the specification. Any reporter or effector molecules can be linked to antibodies using known techniques. The use of such reporter or effector molecules to enhance the properties of antibodies was known to those in the art prior to the filing date of the application.

In view of the foregoing, it is clear from the teachings in the specification and the level of skill in the art that any experimentation necessary to confirm whether a particular encoded protein variant is able to decrease bone mineral content would not rise to the level of undue experimentation as alleged in the rejection. Accordingly, Applicants respectfully request that the rejection under 35 U.S.C. § 112, first paragraph, be withdrawn.

III. The Rejection under 35 U.S.C. §102(e) is moot and should be withdrawn.

The Examiner rejected claims 101 and 103-106 under 35 U.S.C. § 102(e) as allegedly being anticipated by Queen et al. (U.S. Patent No. 6,180,370). The rejection is moot in view

of the amendment to claim 101 made herein, which was made for the reasons discussed above and not for reasons pertaining to patentability. Accordingly, the rejection is moot and should be withdrawn.

IV. Obviousness-type double-patenting rejection.

Claims 88, 89 and 91-100 were rejected under the judicially created doctrine of obviousness-type double patenting in view of claims 1-8 of U.S. Patent No. 6,804,453.

Applicants request that these double patenting rejection be held in abeyance until there is an indication of allowable subject matter. At that time, Applicants will consider filing appropriate disclaimer(s). It is premature to disclaim term before the scope of an allowable claim is clear.

V. Conclusion

In view of the above amendment, Applicants believe the pending application is in condition for allowance. Please charge any deficiency in the fees to Deposit Account No. 13-2855.

Dated: May 27, 2008

Respectfully submitted,

By: /Jeanne M. Brashear/56,301
Jeanne M. Brashear
Registration No.: 56,301
MARSHALL, GERSTEIN & BORUN LLP
233 S. Wacker Drive, Suite 6300
Sears Tower
Chicago, Illinois 60606-6357
(312) 474-6300
Agent for Applicants

APPENDIX A

```
CLUSTAL 2.0.5 multiple sequence alignment


Human-V10I MQLPLALCLICLLVHTAFRVVEGQGWQAFKNDATEIIRELGEYPEPPPELENNKTMNRAE 60
Human-P38R MQLPLALCLVCLLVHTAFRVVEGQGWQAFKNDATEIIRELGEYPEPPPELENNKTMNRAE 60
Human      MQLPLALCLVCLLVHTAFRVVEGQGWQAFKNDATEIIPELGEYPEPPPELENNKTMNRAE 60
Vervet     MQLPLALCLVCLLVHAAFRVVEGQGWQAFKNDATEIIPELGEYPEPPPELENNKTMNRAE 60
Bovine     -----------------------------NDATEIIPELGEYPEPLPEL-NNKTMNRAE 29
Mouse      MQPSLAPCLICLLVHAAFCAVEGQGWQAFRNDATEVIPGLGEYPEPPP--ENNQTMNRAE 58
Rat        MQLSLAPCLACLLVHAAFVAVESQGWQAFKNDATEIIPGLREYPEPPQELENNQTMNRAE 60
                                          *****:*  * *****     **:******

Human-V10I NGGRPPHHPFETKDVSEYSCRELHFTRYVTDGPCRSAKPVTELVCSGQCGPARLLPNAIG 120
Human-P38R NGGRPPHHPFETKDVSEYSCRELHFTRYVTDGPCRSAKPVTELVCSGQCGPARLLPNAIG 120
Human      NGGRPPHHPFETKDVSEYSCRELHFTRYVTDGPCRSAKPVTELVCSGQCGPARLLPNAIG 120
Vervet     NGGRPPHHPFETKDVSEYSCRELHFTRYVTDGPCRSAKPVTELVCSGQCGPARLLPNAIG 120
Bovine     NGGRPPHHPFETKDASEYSCRELHFTRYVTDGPCRSAKPVTELVCSGQCGPARLLPNAIG 89
Mouse      NGGRPPHHPYDAKDVSEYSCRELHYTRFLTDGPCRSAKPVTELVCSGQCGPARLLPNAIG 118
Rat        NGGRPPHHPYDTKDVSEYSCRELHYTRFVTDGPCRSAKPVTELVCSGQCGPARLLPNAIG 120
           *********:::**.*********:**::********************************

Human-V10I RGKWWRPSGPDFRCIPDRYRAQRVQLLCPGGEAPRARKVRLVASCKCKRLTRFHNQSELK 180
Human-P38R RGKWWRPSGPDFRCIPDRYRAQRVQLLCPGGEAPRARKVRLVASCKCKRLTRFHNQSELK 180
Human      RGKWWRPSGPDFRCIPDRYRAQRVQLLCPGGEAPRARKVRLVASCKCKRLTRFHNQSELK 180
Vervet     RGKWWRPSGPDFRCIPDRYRAQRVQLLCPGGAAPRARKVRLVASCKCKRLTRFHNQSELK 180
Bovine     RGKWWRPSGPDFRCIPDRYRAQRVQLLCPGGAAPRARKVRLVASCKCKRLTRFHNQSELK 149
Mouse      RVKWWRPNGPDFRCIPDRYRAQRVQLLCPGGAAPRSRKVRLVASCKCKRLTRFHNQSELK 178
Rat        RVKWWRPNGPDFRCIPDRYRAQRVQLLCPGGAAPRSRKVRLVASCKCKRLTRFHNQSELK 180
           * *****.********************** ***:*************************

Human-V10I DFGTEAARPQKGRKPRPRARSAKANQAELENAY 213
Human-P38R DFGTEAARPQKGRKPRPRARSAKANQAELENAY 213
Human      DFGTEAARPQKGRKPRPRARSAKANQAELENAY 213
Vervet     DFGPEAARPQKGRKPRPRARGAKANQAELENAY 213
Bovine     DFGPEAARPQTGRKLRPRARGTKASRA------ 176
Mouse      DFGPETARPQKGRKPRPGARGAKANQAELENAY 211
Rat        DFGPETARPQKGRKPRPRARGAKANQAELENAY 213
           ***.*:****.*** ** **.:**.:*
```

APPENDIX B

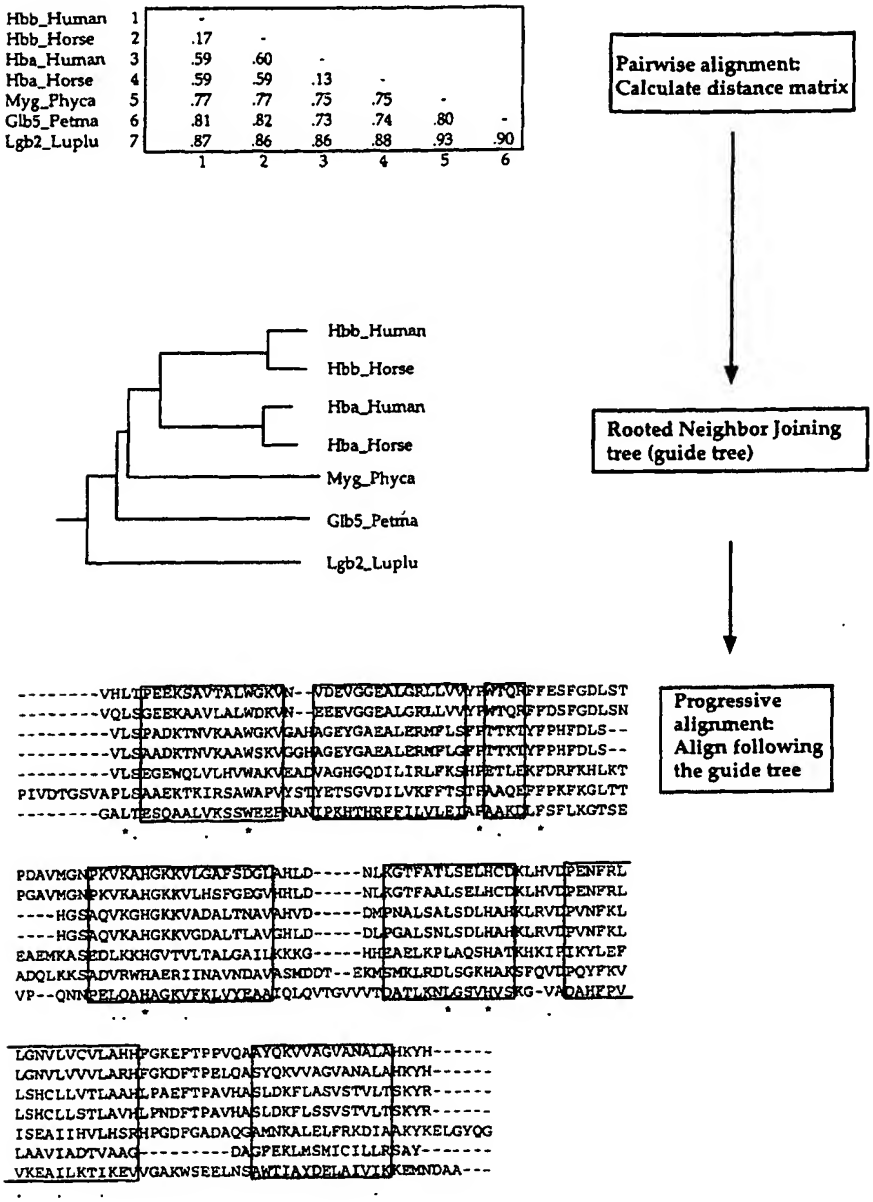
[22] Using CLUSTAL for Multiple Sequence Alignments

By DESMOND G. HIGGINS, JULIE D. THOMPSON, and TOBY J. GIBSON

Introduction

The simultaneous alignment of many nucleotide or amino acid sequences is now one of the commonest tasks in computational molecular biology. It forms a common prelude to phylogenetic analysis of sequences, the prediction of secondary structure (DNA or protein), the detection of homology between newly sequenced genes and existing sequence families, the demonstration of homology in multigene families, and the finding of candidate primers for PCR (polymerase chain reaction). For such a widely required analysis, it is surprising how long it took for practical methods to appear. Up until 1987, it was standard practice to construct multiple alignments manually. This is very tedious and error prone. The basic problem was that direct extensions of the standard dynamic programming approach for the alignment of two sequences were computationally impossible for more than three real sequences. Some methods had been invented[1-4] based on trying to find alignment blocks or on iterating toward a consensus sequence, for example, but these were not very widely used. The first practical methods for the sensitive multiple alignment of many protein sequences appeared in 1987 and 1988[5-10] and were based on an original idea by David Sankoff.[11] The idea is to exploit the fact that groups of sequences are phylogenetically related (if they can be aligned, there is usually an underlying phylogenetic tree). This approach is commonly referred to as progressive alignment.[6] Most of the automatic multiple alignments that appear in the current literature are carried out using this approach.

[1] W. Bains, *Nucleic Acids Res.* **14**, 159 (1986).
[2] E. Sobel and H. M. Martinez, *Nucleic Acids Res.* **14**, 363 (1986).
[3] D. J. Bacon and W. F. Anderson, *J. Mol. Biol.* **191**, 153 (1986).
[4] M. S. Johnson and R. F. Doolittle, *J. Mol. Evol.* **23**, 267 (1986).
[5] W. R. Taylor, *CABIOS* **3**, 81 (1987).
[6] D.-F. Feng and R. F. Doolittle, *J. Mol. Evol.* **25**, 351 (1987).
[7] G. J. Barton and M. J. E. Sternberg, *J. Mol. Biol.* **198**, 327 (1987).
[8] W. R. Taylor, *J. Mol. Evol.* **28**, 161 (1988).
[9] D. G. Higgins and P. M. Sharp, *Gene* **73**, 237 (1988).
[10] F. Corpet, *Nucleic Acids Res.* **16**, 10881 (1988).
[11] D. Sankoff, *SIAM J. Appl. Math.* **78**, 35 (1975).


Hbb_Human 1 -
Hbb_Horse 2 .17 -
Hba_Human 3 .59 .60 -
Hba_Horse 4 .59 .59 .13 -
Myg_Phyca 5 .77 .77 .75 .75 -
Glb5_Petma 6 .81 .82 .73 .74 .80 -
Lgb2_Luplu 7 .87 .86 .86 .88 .93 .90
1 2 3 4 5 6
Pairwise alignment: Calculate distance matrix
Hbb_Human
Hbb_Horse
Hba_Human
Hba_Horse
Myg_Phyca
Glb5_Petma
Lgb2_Luplu
Rooted Neighbor Joining tree (guide tree)
Progressive alignment: Align following the guide tree

Progressive alignment involves making initial guesses as to the phylogenetic relatedness of the sequences and using the branching order in an initial phylogenetic tree to align larger and larger groups of sequences. Start by aligning the most closely related pairs of sequences using dynamic programming and gradually align these groups together, keeping gaps that appear in early alignments fixed. At each stage, align two sequences or one sequence to an existing subalignment or align two subalignments. There are now many variations on the approach. The initial tree may be modified as the procedure progresses[12]; alternative branching orders may be tested at each stage[6]; the initial tree may be replaced by a tree calculated from the fully aligned sequences and the procedure iterated until the alignment (or tree) converges[10]; sequences may be aligned together in a nonphylogenetic manner but rather by adding sequences one at a time to a growing alignment.[7] Further, one can use different strategies for aligning the groups of prealigned sequences. For example, one can base the alignment only on the alignment of the two most closely related sequences (one from each alignment).[5] In most current implementations, the subalignments are aligned together using information from all of the constituent sequences with an extension of the profile alignment approach.[13] An example of the approach is shown in Fig. 1 for the alignment of seven globin sequences of known tertiary structure.

Algorithmically, the progressive approach is considered to be "heuristic" insofar as the method is not guaranteed to produce alignments with any particular mathematical property such as maximum alignment score. The method is, however, soundly based biologically and has the great

[12] J. Hein, *Mol. Biol. Evol.* **6**, 649 (1989).

[13] M. Gribskov, A. D. McLachlan, and D. Eisenberg, *Proc. Natl. Acad. Sci. U.S.A.* **84**, 4355 (1987).

FIG. 1. Outline of the progressive multiple alignment approach in CLUSTAL W. Seven globin sequences of known tertiary structure are used (the sequence names are identifiers from the SWISS-PROT sequence database). The alignment was carried out using default parameters except that the Dayhoff PAM weight matrices [M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt, *in* "Atlas of Protein Sequence and Structure" (M. O. Dayhoff, ed.), Vol. 5, Suppl. 3, p. 345. National Biomedical Research Foundation, Washington, D.C., 1978] were used instead of BLOSUM [S. Henikoff and J. G. Henikoff, *Proc. Natl. Acad. Sci. U.S.A.* **89**, 10915 (1992)]. The approximate positions of the seven α helices, common to all sequences, are shown as boxes. The alignment is carried out by aligning the two α-globins (Hba_Human and Hba_Horse) together; then the two β-globins are aligned (Hbb_Human and Hbb_Horse); then the two aligned α-globins are aligned to the two aligned β-globins; finally, the whale myoglobin (Myg_Phyca), the lamprey cyanohemoglobin (Glb5_Petma), and the lupine leghemoglobin (Lgb2_Luplu) are aligned one at a time to the growing alignment.

advantage of speed and simplicity. One can align hundreds of sequences, even on personal computers. More importantly, the sensitivity of the approach, as judged by the ability to align distantly related sequences, is very high. In simple cases, it is common to derive alignments which are impossible to improve by eye. In these cases, the method can be said to be a satisfactory replacement for manual alignment. In more difficult cases, the method usually gives a useful starting point for further refinement. In this chapter, we describe some of the problems of progressive multiple alignment and some simple modifications which, we believe, greatly improve the sensitivity for difficult protein alignments. All of the methods described here are freely available in a computer program called CLUSTAL W which can be run under a wide variety of operating systems.

Problems with Progressive Alignment

There are two obvious and interrelated problems inherent in the progressive alignment approach: (1) the local minimum problem and (2) the parameter choice problem. The local minimum problem stems from the "greedy" nature of the algorithm. Every time an alignment is carried out, some proportion of the residues will be misaligned. This proportion will be very small (or nonexistent) for very closely related proteins (e.g., at least 50% identical) but will increase as more and more divergent sequences are used. Any mistakes that appear during early alignments in a progressive multiple alignment cannot be corrected later as new sequence information is added. If the data set contains sequences of different degrees of divergence, the first alignments may be very accurate, and by the time the most diverged sequences are aligned, some information about gap frequency and residue conservation at each position will be available. In this case, the progressive approach may work very well. In other cases, however, if the first alignments are not correct they cannot be corrected later.

It is commonly argued that the local minimum problems stems largely from errors in the branching order of the initial phylogenetic tree. Consequently, many authors have devoted great effort to investigating alternative topologies or advocating particular methods of phylogenetic analysis. For example, previous versions of CLUSTAL[9,14,15] have been criticized[16] for using UPGMA[17] to generate initial trees as UPGMA is notorious for giving

[14] D. G. Higgins and P. M. Sharp, *CABIOS* **5**, 151 (1989).
[15] D. G. Higgins, A. J. Bleasby, and R. Fuchs, *CABIOS* **8**, 189 (1992).
[16] C.-B. Stewart, *Nature (London)* **367**, 26 (1994).
[17] P. H. A. Sneath and R. R. Sokal, "Numerical Taxonomy." Freeman, San Francisco, 1973.

incorrect branching orders when rates of substitution vary greatly in different lineages. This criticism is erroneous. If the alignment is simple enough, almost any tree will give the correct alignment. If the alignment is sufficiently difficult, almost any tree will give the wrong alignment. There is no one-to-one correspondence between having the correct tree topology and getting the right alignment. The better the tree, then the better the chances of getting a good alignment, but there are no guarantees. Even if UPGMA is not ideal for obtaining correct tree topologies, it can be argued that it is still very useful for alignment purposes. At each stage in the alignment process, align the most similar remaining sequences or subalignments so as to minimize the alignment errors at that step. UPGMA gives this property automatically. Nonetheless, we now provide the neighbor-joining method[18] for making initial trees because it seems to provide more reliable tree topologies and gives better estimates of tree branch lengths which we use to weight sequences and adjust the alignment parameters dynamically.

The local minimum problem is intrinsic to progressive alignment, and we do not provide any direct solutions. The only way to correct it is to use an overall measure of multiple alignment quality and find the alignment which maximizes this measure. This can be done directly for small numbers of sequences using the program MSA[19] but is, for now, uncomputable for more than about seven sequences. MSA computes approximate bounds for the location of the best pathway through the N-dimensional (for N sequences) dynamic programming array and then carries out full dynamic programming in this restricted area. It may be possible to use stochastic or iterative optimization procedures[20,21] to optimize multiple alignments of many sequences in the future. Even if practical solutions are found, we believe that the parameter choice problem, described below, is just as important and will still preclude high accuracy alignments if it is not addressed.

The parameter choice problem stems from using just one set of parameters (normally an amino acid substitution matrix and two gap penalties) and hoping that these will be appropriate over all parts of all the sequences to be aligned. If the sequences are very similar, almost any reasonable parameters will give a good alignment. For highly divergent sequences, however, the exact choice of parameters may have a great effect on align-

[18] N. Saitou and M. Nei, *Mol. Biol. Evol.* **4**, 406 (1987).
[19] D. Lipman, S. F. Altschul, and Kececioglu, *J. Proc. Natl. Acad. Sci. U.S.A.* **86**, 4412 (1989).
[20] O. Gotoh, *CABIOS* **9**, 361 (1993).
[21] C. E. Lawrence, S. F. Altschul, M. S. Boguski, J. S. Liu, A. F. Neuwald, and J. C. Wooton, *Science* **262**, 208 (1993).

ment quality. Different weight matrices from the well-known PAM[22] or BLOSUM[23] series are appropriate for aligning sequences of different evolutionary distances. For very similar sequences, even an identity matrix will provide sensible alignments, but for sequences in the so-called twilight zone[24] (sequences of roughly 20–25% identity), the exact values given to each type of substitution may be critical. Further, there is no particular reason why the same gap penalties should work equally well at all positions in an alignment. Gaps tend to occur far more often between the main secondary structure elements of α helices and β strands than within.[25] With the latest CLUSTAL program (CLUSTAL W), we attempt to attack this problem by computing position-specific gap opening and extension penalties as the alignment proceeds. We also use different amino acid weight matrices; "hard" ones for closely related sequences and "softer" ones for more divergent sequences.

We believe that the correct use of parameters at different positions is important. We provide simple heuristic methods for doing this which seem to work well in difficult test cases. Other authors have attacked the problem using more systematic methods such as hidden Markov models[26] or have exploited structural information when the structure of one or more of the sequences is known.[27]

CLUSTAL W

CLUSTAL W[28] is derived directly from the CLUSTAL[9,14] and CLUSTAL V series of programs. The "W" in the name stands for "weighting" as we now give different weights to sequences and parameters at different positions in alignments. The main feature of the old programs was the ability to align many sequences quickly, even on a personal computer, with a minimal sacrifice in sensitivity. The new program offers similar speed as before (CLUSTAL W is in fact slower than the older programs, but, fortunately, advances in the power of personal computers and workstations have canceled this out) but provides a number of new features and appears

[22] M. O. Dayhoff, R. M. Schwartz, and B. C. Orcutt, *in* "Atlas of Protein Sequence and Structure" (M. O. Dayhoff, ed.), Vol. 5, Suppl. 3, p. 345. National Biomedical Research Foundation, Washington, D.C., 1978.

[23] S. Henikoff and J. G. Henikoff, *Proc. Natl. Acad. Sci. U.S.A.* **89**, 10915 (1992).

[24] R. F. Doolittle, "URFs and ORFs: A Primer on How to Analyze Derived Amino Acid Sequences." University Science Books, Mill Valley, California, 1987.

[25] S. Pascarella and P. Argos, *J. Mol. Biol.* **224**, 461 (1992).

[26] A. Krogh, M. Brown, S. Mian, K. Sjölander, and D. Haussler, *J. Mol. Biol.* **235**, 1501 (1994).

[27] G. J. Barton and M. J. E. Sternberg, *Protein Eng.* **1**, 89 (1987).

[28] J. D. Thompson, D. G. Higgins, and T. J. Gibson, *Nucleic Acids Res.* **22**, 4673 (1994).

to be more sensitive for difficult protein alignments. The main new features are (1) support for more file formats for trees, sequence data sets, and alignments; (2) optional, full dynamic programming alignments for estimating the initial pairwise distances between all the sequences; (3) neighbor-joining[18] trees for the initial guide trees, used to guide the progressive alignments; (4) sequence weighting to correct for unequal sampling of sequences at different evolutionary distances; (5) dynamic calculation of sequence- and position-specific gap penalties as the alignment procedes; (6) the use of different weight matrices for different alignments; and (7) improved facilities for adding new sequences to an existing alignment.

The source code of CLUSTAL W, version 1.5 (April 1995), is available free of charge from the EMBL E-mail file server or by anonymous ftp to the EMBL ftp server. Compiled versions are also available for MS-DOS and Macintosh computers. With the Macintosh version, we also supply the NJplot program of Manolo Gouy (University of Lyon) which allows for the graphical display and manipulation of phylogenetic trees.

Use ftp to connect to ftp.ebi.ac.uk and give user name anonymous and full E-mail address as password. The four versions are as follows:

/pub/software/vax/clusalw.uue	uuencoded ZIP archive for VMS
/pub/software/unix/clustalw.tar.Z	compressed tar archive for UNIX
/pub/software/dos/clustal$.exe	self-extracting archive for MS-DOS
/pub/software/mac/clustalw.sea.hqx	Binhex encoded self-extracting archive

The MS-DOS and UNIX versions should be transferred in binary mode; the VMS and Mac versions in ASCII.

Position-Specific Gap Penalties

Here we give a brief summary of the methods used to calculate position-specific gap penalties for protein alignments. Two gap penalties are used initially: a gap opening penalty (GOP) and a gap extension penalty (GEP). Traditionally, one will choose values for these parameters before alignment and use the same values for all sequences. The exact values used are usually the default values offered by the software, which are often chosen empirically by the software authors by trial and error for a given amino acid weight matrix. In a simple world where one knew the positions of all secondary structure elements (α helices and β strands) in all or some of the sequences, one could increase the GOP (and GEP) at each position inside a helix or strand and decrease it between them.[27] This would force gaps to occur most often in loop regions, which is what is observed in

practice with test cases from protein structure superposition.[25] One could be more sophisticated and make the GOP highest at the center of helices and strands and reduce it at the edge, allowing some gaps to occur at the ends of secondary structure elements. If one does not know the secondary structure of the sequences, as is normally the case, this is not possible. In CLUSTAL W, we use a set of very simple rules to help modify the GOP and GEP at each position in a sequence or prealigned group of sequences, depending on the residues that occur at each position and the frequency of gaps at each position. These rules are simple heuristics that seem to work very well in practice, although it should be possible to derive similar rules with greater statistical or mathematical validity.

Before any two sequences or prealigned groups of sequences are aligned, we calculate initial values for the GOP and GEP as functions of the amino acid weight matrix to be used, the sequence (or alignment) lengths, and the divergence between the sequences. The values for GOP and GEP are set from a user-controlled menu (defaults are offered) and then modified as follows:

$$\text{GOP} \rightarrow A * B * \{\text{GOP} + \log[\min(N, M)]\} \quad (1)$$

where N and M are the lengths of the sequences to be aligned, A is the average value for a mismatch in the amino acid weight matrix, and B is the percent identity of the two sequences. The GEP is then modified using the following formula:

$$\text{GEP} \rightarrow \text{GEP} * [1.0 + |\log(N/M)|] \quad (2)$$

where N and M are, again, the lengths of the two sequences.

The overall effect of these transformations is to allow for the use of different weight matrices with sequences of different degrees of divergence and to try to correct for some side effects of using sequences of different lengths. If the sequences are greatly different in length (as measured by the ratio N/M), the GEP is increased to try to inhibit the appearance of too many long gaps in the shorter sequence.

Next, tables of GOP and GEP values are calculated for each of the two sequences or groups of sequences to be aligned, one GOP and GEP for each position. Initially, these values are all the same, as calculated using Eqs. (1) and (2) above. These are then modified at each position using four rules. The overall aim is to encourage gaps to occur in likely loop regions. Informally, the rules are (1) use lower gap penalties at positions where gaps already occur; (2) increase gap penalties adjacent to positions where gaps already occur; (3) reduce gap penalties where stretches of hydrophilic residues occur; and (4) increase or decrease gap penalties using tables of the observed frequencies of gaps adjacent to each of the 20 amino acids.[25]

If there are gaps at a position in a group of prealigned sequences (this rule and the following one do not apply to single sequences), then the GOP is reduced in proportion to the number of sequences with a gap at that position and the GEP is lowered by one-half. The new GOP is calculated as

$$\text{GOP} \rightarrow \text{GOP} * 0.3 * (W/N) \quad (3)$$

where W is the number of sequences without a gap at the position and N is the number of sequences.

If a position contains no gaps but is within eight residues of an existing gap (this value of 8 can be changed from a menu), the GOP is increased as follows:

$$\text{GOP} \rightarrow \text{GOP} * \{2 + [8 - (D) * 2]/8\} \quad (4)$$

where D is the distance from the gap.

A run of five (this number can be changed from a menu) consecutive, hydrophilic residues is considered to be a hydrophilic stretch. The residues that are considered to be hydrophilic are conservatively set to D, E, G, K, N, Q, P, R, and S by default but can be changed by the user. Any positions with no gaps that are spanned by such a stretch of residues get the GOP reduced by one-third.

In Table I, we list 20 residue-specific gap propensity values. These are derived from the observed frequencies of gaps adjacent to each residue in

TABLE I
RESIDUE-SPECIFIC GAP OPENING PENALTY FACTORS[a]

Residue	Penalty	Residue	Penalty
A	1.13	M	1.29
C	1.13	N	0.63
D	0.96	P	0.74
E	1.31	Q	1.07
F	1.20	R	0.72
G	0.61	S	0.76
H	1.00	T	0.89
I	1.32	V	1.25
K	0.96	Y	1.00
L	1.21	W	1.23

[a] These values are derived from the observed frequencies of gaps adjacent to each residue in alignments of sequences of known tertiary structure.[25] The values were transformed from the published values such that the bigger the number, the less likely a gap is to occur adjacent to that residue. The numbers are then used as simple multiplication factors to modify gap opening penalties, normalized around a value of 1.0 for histidine.

alignments of sequences with known three-dimensional structure.[25] If a position does not contain a gap or a hydrophilic stretch, then the values in Table I are used as simple multiplication factors to increase or decrease the GOP; for example, a position with only glycine will get a reduced GOP (multiplied by 0.61), whereas a position with only methionine will get an increased GOP (multiplied by 1.29). If there is a mixture of residues at a position, then the multiplication factor is the average of those for each residue, one from each sequence.

The overall effect of the four rules can be seen in Fig. 2 on a small stretch of alignment from four globin sequences. The GOP is highest adjacent to a gap and lowest at a gap position or where hydrophilic runs occur. These rules are most useful when there are already some sequences correctly aligned. Then, new gaps will tend to concentrate in areas where gaps already occur and will promote a blocklike appearance in the final alignment. For the first alignments, before any sequences are aligned, there are no gaps yet and the first two rules above cannot be used; only the residue-specific gap frequency rule and hydrophilic stretch rules can be used.

Sequence Weighting

In most real data sets of protein sequences, it is common to have unequal sampling of sequences at different evolutionary distances. Frequently, there

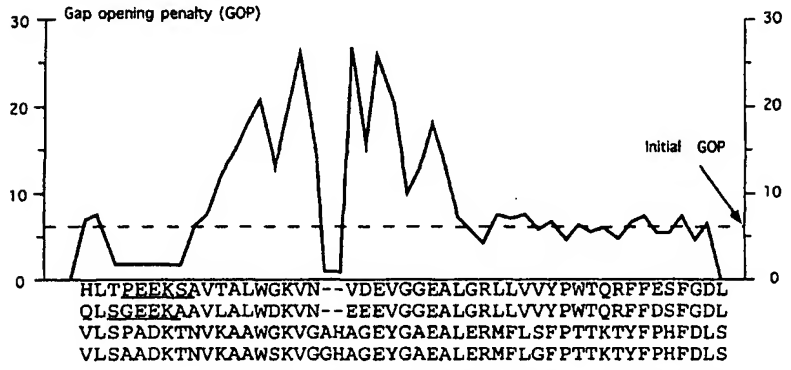


FIG. 2. Illustration of the effect of modifying the gap opening penalty (GOP) on a stretch of globin alignment from Fig. 1. The initial GOP is shown as a dotted line, and the position-specific GOP values are plotted along the alignment. The GOP is lowest at positions with gaps and where hydrophilic stretches occur (two such stretches are underlined). The GOP is highest within eight residues of gaps. The rest of the variation is caused by the residue-specific factors from Table I.

are clusters of closely related sequences and small numbers of highly diverged ones. For example, in the data set illustrated in Fig. 1, there are two mammalian α-globins and two β-globins. These pairs are almost identical and provide little extra information for alignment purposes. Traditionally, during multiple alignment or when using aligned sequences as profiles for database searching, one gives equal weight to all sequences. In the globin example, this means that the two α-globins are given as much weight each as the more diverged leghemoglobin. Several authors have addressed the problem of sequence weighting in order to correct for this effect[29]; closely related sequences are down-weighted, while relatively more distant ones receive greater weight. In profile searches, this has been shown to increase the sensitivity of the search as measured by the ability to detect distant relatives of the sequences in the profile.[30-32] Sequence weights were first used for multiple alignments by Vingron and Argos[33] and are also used in the MSA[19] program.

There are several methods available for sequence weighting. For our purposes, it must be possible to derive weights for unaligned sequences as the weights will be used to help arrive at the multiple alignment itself. This prevents the use of methods that use multiple alignments as input. We use a simple tree-based method that only requires a tree describing the rough relatedness of the sequences.[30] The guide trees, used to guide the multiple alignment, provide this. The method requires a rooted tree with branch lengths. As a first approximation, the weights are calculated as the distance of each sequence from the root of the tree. This gives increased weight to the most diverged sequences and less to the more conserved ones. Second, if two sequences (or groups of sequences) share a common internal branch, the length of the internal branch is shared when deriving the weights. This automatically downweights related sequences in proportion to the degree of relatedness. An example is shown in Fig. 3, using the guide tree from Fig. 1. Here the leghemoglobin (Lgb2_Luplu) gets a weight of 0.442, which is equal to the length of the branch from the root to it. The human α-globin (Hba_Human) receives a weight equal to the length of branch leading to it that is not shared by any other sequences (0.055) plus one-half the length of the branch shared with the horse α-globin (0.219/2) plus one-quarter the length of the branch shared by all four hemoglobins (0.061/4) plus one-fifth the branch shared between the hemoglobins and the myo-

[29] M. Vingron and P. R. Sibbald, *Proc. Natl. Acad. Sci. U.S.A.* **90**, 8777 (1993).
[30] J. D. Thompson, D. G. Higgins, and T. J. Gibson, *CABIOS* **10**, 19 (1994).
[31] R. Lüthy, I. Xenarios, and P. Bucher, *Protein Sci.* **3**, 139 (1994).
[32] S. Henikoff and J. G. Henikoff, *J. Mol. Biol.* **243**, 574 (1994).
[33] M. Vingron and P. Argos, *CABIOS* **5**, 115 (1989).

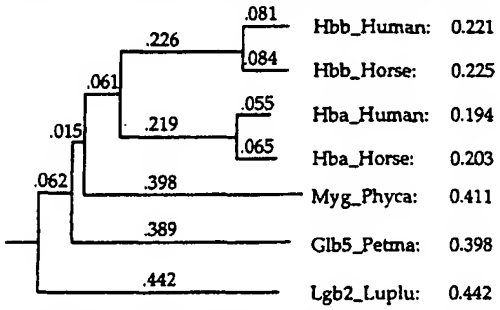


FIG. 3. Sequence weights for the seven globin sequences from Fig. 1. A rooted neighbor-joining tree is shown with branch lengths. The weights are shown for each sequence before normalization (the weights are normalized so as to make the largest equal to 1.0).

globin (0.015/5) plus one-sixth the branch shared by all the vertebrate globins (0.062/6). This gives a total weight of 0.194 before normalization. All weights are normalized to make the largest weight equal to one.

These weights are intuitive, simple to calculate, and have been shown to be useful in increasing the sensitivity of profile searches.[30,32] With multiple alignment, they are used to give different weight to the contributions of different sequences to the alignment scores when aligning two subalignments or when aligning a sequence to a subalignment. The weights are used as simple multiplication factors when calculating the alignment score between two positions.

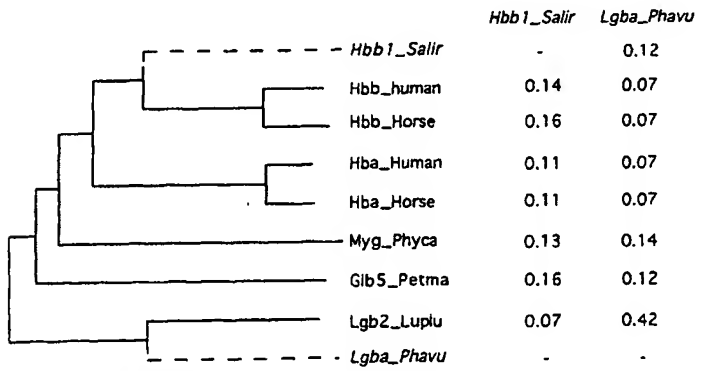
Weights for Adding New Sequences to Existing Alignment

Sequence weights are also useful when adding new sequences to an existing alignment. In CLUSTAL W, we provide facilities to do this in three ways: (1) add a single sequence to an alignment; (2) add a set of new sequences one at a time to an alignment; (3) align two existing alignments. With methods 1 and 2, there is a further use of sequence weighting. If the sequence to be aligned is much closer to some of the sequences in the alignment (the new sequence can be said to go "inside" the underlying tree), then one can exploit this to give extra weight to the most closely related examples (most closely related to the new sequence). This will help to make sure that the placement of new gaps is most influenced by the close relatives rather than distantly related sequences. The weights that achieve this effect are, again, very simple to derive. Pairwise alignments and resulting simple alignment distances (mean number of differences per site, ignoring gaps and not corrected for multiple hits) are calculated be-

tween the new sequence and each sequence in the old alignment. Then, new weights are calculated for each sequence in the alignment as 1.0 minus the distance from the new sequence. This has the effect of giving a weight of 1.0 to identical sequences (identical to the new sequence) and a low weight to distant relatives. Finally, these weights are multiplied with the original tree weights for each sequence in order to combine the properties of the two types of weights, and the weights are normalized so as to sum to 1.

An example is shown in Fig. 4 where two globins are added to the previous globin alignment. The two sequences are compared to each of the sequences in the alignment. The sequences are added to the alignment such that the most similar ones (most similar, on average, to the sequences in the alignment) are added first. In this case, the trout β-globin (Hbb1_Salir) is added first using the weights shown in Fig. 4. These weights give increased weight to the most closely related sequences (closest to the new sequence) but also downweight sequences that are closely related to each other. Then, a new set of weights are calculated for the addition of the new leghemoglobin sequence (Lgba_Phavu), including a weight for the trout β-globin.

We have not examined the properties of these new weights for adding sequences to an alignment in any detail or carried out extensive empirical



	Hbb1_Salir	*Lgba_Phavu*
Hbb1_Salir	-	0.12
Hbb_human	0.14	0.07
Hbb_Horse	0.16	0.07
Hba_Human	0.11	0.07
Hba_Horse	0.11	0.07
Myg_Phyca	0.13	0.14
Glb5_Petma	0.16	0.12
Lgb2_Lupiu	0.07	0.42
Lgba_Phavu	-	-

FIG. 4. Normalized weights for the addition of a trout β-globin (Hbb1_Salir) and a bean leghemoglobin (Lgba_Phavu) to the alignment of seven globin sequences in Fig. 1. The positions in the tree for the new sequences are shown with dashed lines. First, the trout β-globin is added, and the seven weights needed for this are shown. Then the bean globin is added, and eight weights are needed for this (one weight for each of the original seven globins and one for the trout globin). The weights have the effect of simultaneously upweighting sequences that are similar to the new sequence and downweighting sequences that have other close relatives in the tree. The weights are normalized to sum to 1.0.

evaluations. They are, however, intuitive and simple to calculate. It seems to us to be of great importance to make use of such weights or to develop and evaluate new weighting schemes as more and more databases of large alignments are created that need to be updated automatically. These weights help to exploit the information contained in large alignments. An alignment of a few hundred or more sequences that has been carefully created by experts, sometimes incorporating structural information, is an important and useful resource which contains much information to help characterize new sequences.

Phylogenetic Trees

All trees in CLUSTAL W are calculated using the neighbor-joining method.[18] These are used as guide trees to guide the multiple alignments or can be produced after multiple alignment with a bootstrap[34] option. This is a distance matrix approach which is fast to calculate but which gives the correct tree topology in a wide variety of situations. The method produces unrooted trees with estimates of branch lengths for each branch.

For the calculation of guide trees, the user can choose between fast approximate pairwise alignment of sequences using a *k*-tuple approach[35] based only on identities or full dynamic programming[36] with a weight matrix and two gap penalties. The former are extremely fast to calculate and are useful if the user has hundreds of sequences, but the latter are more accurate. Distances are calculated as the number of exact matches in the best alignment between the pair of sequences divided by the number of positions considered, ignoring positions with gaps. With the guide trees, there is no correction for multiple substitutions. The distances are given to the neighbor-joining method, and an unrooted guide tree is produced. The tree is rooted by a "midpoint" approach.[30] Root placement involves making the mean distance from the root to the tips of the tree equal on both sides of the root. The biological validity of this method of placing the root depends on the quality of the "molecular clock" in the data set. For the current application, however, it does not matter if the root is incorrectly placed. This is because placing the root using the midpoint method is the equivalent of always aligning the next most closely related sequences or groups of sequences on the tree, ending when all sequences are aligned. This is the desired behavior.

After multiple alignment (or after reading a full multiple alignment

[34] J. Felsenstein, *Evolution* **39**, 783 (1985).
[35] W. J. Wilbur and D. J. Lipman, *Proc. Natl. Acad. Sci. U.S.A.* **80**, 726 (1983).
[36] S. B. Needleman and C. Wunsch, *J. Mol. Biol.* **48**, 444 (1970).

from a file), more accurate trees can be calculated using distances calculated from the fully aligned sequences. For each distance calculation, gaps are not considered, but there is an option to ignore all sites where a gap occurs in any of the sequences. Although this sounds wasteful of data (it removes any site where a gap occurs) it does have the advantage of basing all distance calculations on the same number of sites. For sequences of similar length, this makes little difference. It has the added advantage of automatically removing the most ambiguous sites from the alignment (those that are hardest to align and where the exact alignment may be arbitrary or an artifact of the alignment process). There is a second option to correct the distances for multiple substitutions. This has little effect on small distances (e.g., sequences more than 80% identical) but will stretch large distances considerably to compensate for the great number of hidden substitutions that are estimated to have occurred. For example, using the Dayhoff model of protein evolution,[22] if one observes two amino acid sequences that are 20% identical (distance of 0.80 differences per site), one estimates that 250 substitutions have occurred per 100 sites (distance of 2.5 substitutions per site). Gaps are not considered in these distance calculations for two reasons. First, it is problematic how to score gaps, and, second, it is not known whether gaps appear and disappear in a clocklike manner. For many protein sequences, it appears that substitutions occur in a reasonably regular manner over time (the so-called molecular clock hypothesis), and this allows one to use sequences to derive phylogenetic information. There may not be an equivalent "gap clock," and if there is it is not known how to calibrate it.

To correct simple protein distances (number of observed differences per site) for multiple hits, it is common practice to use a formula from Kimura[37]:

$$K_{aa} = -\ln(1 - D - D^2/5) \tag{5}$$

where K_{aa} is the estimated, corrected distance and D is the observed distance. This formula is a curve-fitting approximation to a table of corrected distances derived from the Dayhoff model of protein evolution.[22] The formula is simple to calculate and is an extremely accurate fit to the Dayhoff model in the range 0.0 to 0.75 observed distance. Above this, however, the approximation becomes inaccurate, and at D above 0.85 or so, the formula cannot be evaluated as it requires finding the logarithm of a negative number. With CLUSTAL W, we use the Kimura formula for D up to 0.75 and use precalculated tables of corrected distances for all D above this in intervals of 0.001. These tables were calculated using the Dayhoff model

[37] M. Kimura, "The Neutral Theory of Molecular Evolution." Cambridge Univ. Press, Cambridge, 1983.

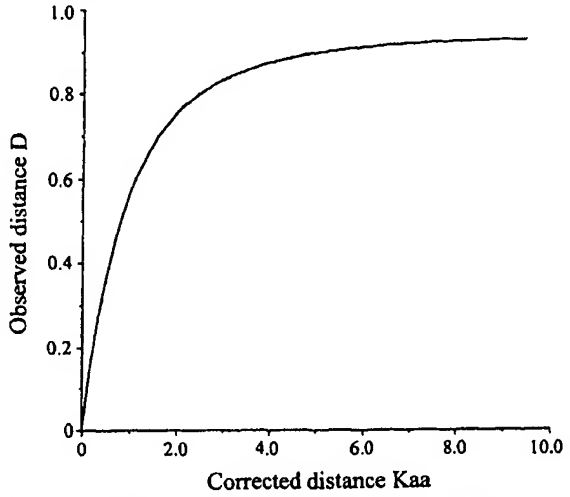


FIG. 5. Relationship between the simple observed distance between two protein sequences (mean number of differences per site) and the estimated corrected distance (the number of substitutions per site that have occurred, including multiple hits). These values were calculated using the Dayhoff model of amino acid substitution[22] and are hard coded in CLUSTAL W.

of amino acid substitution, from which the well-known PAM tables of amino acid similarity are derived. The graph in Fig. 5 shows the relationship between K_{aa} and D. These values were calculated using the Dayhoff model and hard coded into CLUSTAL W.

The Dayhoff model cannot correct any observed distances greater than 0.93 or so as the model reaches equilibrium at this point. For idealized proteins that have reached a divergence of 0.93 observed differences per site, substitutions are just as likely to restore an identity at a site as remove one. Therefore, we correct any observed distances above 0.93 to the arbitrary level of 10.0. This is crude, but such distances are rare with real sequences. If the sequences are this divergent, then alignment is very difficult to begin with and any tree making methods will have great problems making sensible trees. During bootstrapping, however, such distances can occur randomly, even if no sequences are as diverged as this. With CLUSTAL W, the user is warned if this occurs. Provided it occurs rarely, it seems to have no detrimental effect on the bootstrap results. If it becomes critical, however, users are advised to use the more sophisticated distance calculation method provided in the PROTDIST program of the PHYLIP pack-

age.[38] This method uses a Dayhoff model to estimate numbers of substitutions per site, but does so taking all mismatches into account as well as identities, and will be appropriate for sequences of any amino acid composition.

For graphical display of trees, we recommend that users use the DRAWGRAM and DRAWTREE programs of the PHYLIP package[38] or TREETOOL,[39] if the user has access to a SUN computer. For Macintosh computers, we provide the extremely useful and simple to use NJPLOT program of Manolo Gouy.[40] NJPLOT reads trees in the widely used New Hampshire format and allows the user to make simple manipulations and save the tree as PICT format, which can then be used in many Macintosh drawing and graphics packages. The bootstrap trees produced by CLUSTAL W include the bootstrap support levels as extra labels. These are displayed by both TREETOOL and NJPLOT.

Summary

We have tested CLUSTAL W in a wide variety of situations, and it is capable of handling some very difficult protein alignment problems. If the data set consists of enough closely related sequences so that the first alignments are accurate, then CLUSTAL W will usually find an alignment that is very close to ideal. Problems can still occur if the data set includes sequences of greatly different lengths or if some sequences include long regions that are impossible to align with the rest of the data set. Trying to balance the need for long insertions and deletions in some alignments with the need to avoid them in others is still a problem. The default values for our parameters were tested empirically using test cases of sets of globular proteins where some information as to the correct alignment was available. The parameter values may not be very appropriate with nonglobular proteins.

We have argued that using one weight matrix and two gap penalties is too simplistic to be of general use in the most difficult cases. We have replaced these parameters with a large number of new parameters designed primarily to help encourage gaps in loop regions. Although these new parameters are largely heuristic in nature, they perform surprisingly well and are simple to implement. The underlying speed of the progressive

[38] J. Felsenstein, *Cladistics* **5**, 164 (1989).

[39] M. Maciukenas, University of Illinois, USA, unpublished. Available by anonymous ftp from rdp.life.uiuc.edu (/pub/RDP/programs/TreeTool).

[40] M. Gouy, University of Lyon, France, unpublished. Available for Apple Macintosh computers by anonymous ftp frpm ftp.ebi.ac.uk (/pub/software/mac/NJplot.sea.hqx).

alignment approach is not adversely affected. The disadvantage is that the parameter space is now huge; the number of possible combinations of parameters is more than can easily be examined by hand. We justify this by asking the user to treat CLUSTAL W as a data exploration tool rather than as a definitive analysis method. It is not sensible to automatically derive multiple alignments and to trust particular algorithms as being capable of always getting the correct answer. One must examine the alignments closely, especially in conjunction with the underlying phylogenetic tree (or estimate of it) and try varying some of the parameters. Outliers (sequences that have no close relatives) should be aligned carefully, as should fragments of sequences. The program will automatically delay the alignment of any sequences that are less than 40% identical to any others until all other sequences are aligned, but this can be set from a menu by the user. It may be useful to build up an alignment of closely related sequences first and to then add in the more distant relatives one at a time or in batches, using the profile alignments and weighting scheme described earlier and perhaps using a variety of parameter settings.

We give one example using SH2 domains. SH2 domains are widespread in eukaryotic signalling proteins where they function in the recognition of phosphotyrosine-containing peptides.[41] In the chapter by Bork and Gibson ([11], this volume), Blast and pattern/profile searches were used to extract the set of known SH2 domains and to search for new members. (Profiles used in database searches are conceptually very similar to the profiles used in CLUSTAL W: see the chapters [11] and [13] for profile search methods.) The profile searches detected SH2 domains in the JAK family of protein tyrosine kinases,[42] which were thought not to contain SH2 domains. Although the JAK family SH2 domains are rather divergent, they have the necessary core structural residues as well as the critical positively charged residue that binds phosphotyrosine, leaving no doubt that they are bona fide SH2 domains.

The five new JAK family SH2 domains were added sequentially to the existing alignment of 65 SH2 domains using the CLUSTAL W profile alignment option. Figure 6 shows part of the resulting alignment. Despite their divergent sequences, the new SH2 domains have been aligned nearly perfectly with the old set. No insertions were placed in the original SH2 domains. In this example, the profile alignment procedure has produced better results than a one-step full alignment of all 70 SH2 domains, and in

[41] I. Sadowski, J. C. Stone, and T. Pawson, *Mol. Cell. Biol.* **6**, 4396 (1986).
[42] A. F. Wilks, A. G. Harpur, R. R. Kurban, S. J. Ralph, G. Zuercher, and A. Ziemiecki, *Mol. Cell. Biol.* **11**, 2057 (1991).

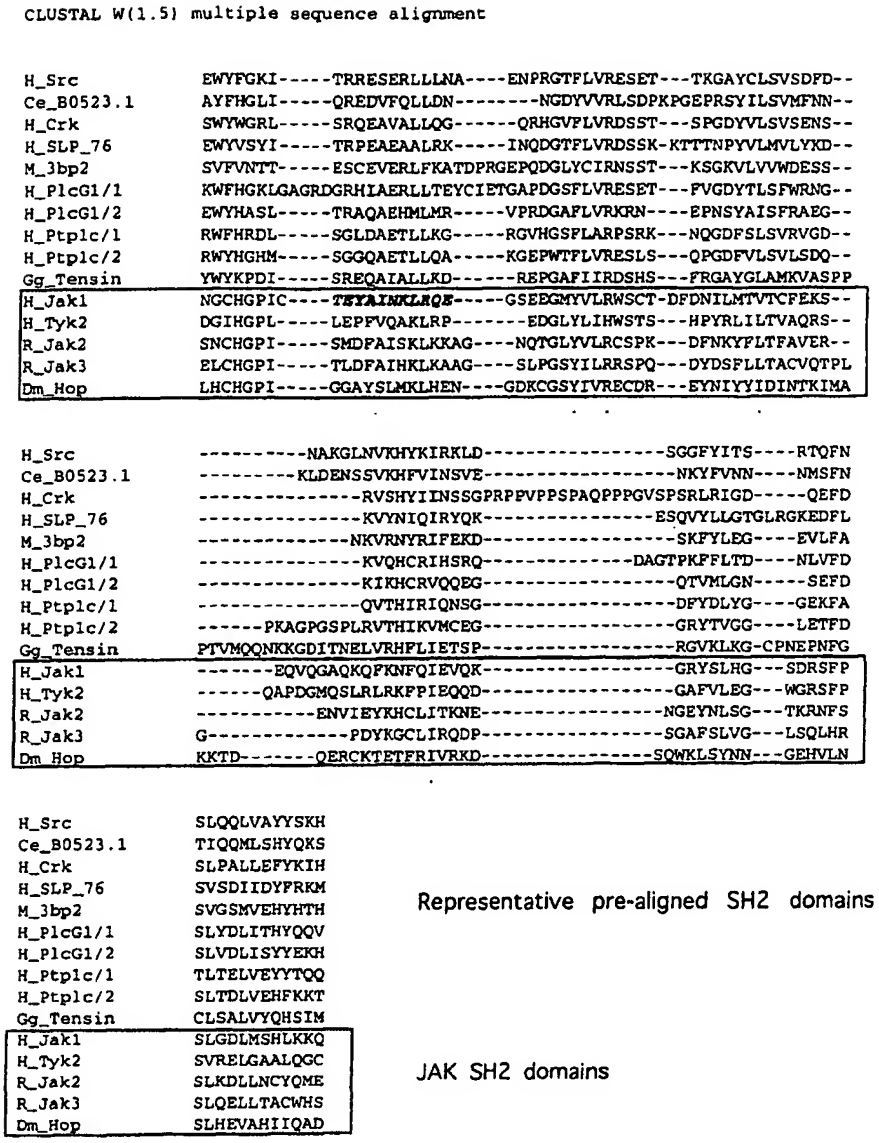


FIG. 6. Profile alignment adding sequentially five newly detected JAK SH2 domains to an existing SH2 alignment. Because of space restrictions, just 10 of the 65 original SH2 domains are shown. All 65 domains were actually used for the profile alignment. The sole error occurs in H_Jak1 block 2 (shown in boldface italics) which is misaligned one residue leftward.

considerably less time. In this example, it is roughly five times faster to add the new sequences one at a time to the existing SH2 alignment than it is to recalculate the full alignment. It is also more accurate and gives the user greater control.

[23] Combined DNA and Protein Alignment

By JOTUN HEIN and JENS STØVLBÆK

Introduction

Most long DNA sequences contain coding regions, and thus it is optimal to use information from both the DNA sequence and the coded protein when comparing such a sequence to a homologous variant. In this chapter we present a heuristic algorithm that can compare DNA with both coding and noncoding regions, but also multiple reading frames, and determine which exons are homologous. A program, GenAl (genomic alignment), has been developed that implements the algorithm. It is demonstrated by comparing HIV2 (human immunodeficiency virus type 2) with HIV1. A stochastic model of the evolution of the complete virus has also been developed that allows estimation of the amount of selective constraint on different genes (including overlapping regions), the equilibrium base composition, and lastly the transversion and transition distances. This method has been applied to two HIV2's.

Comparisons of longer genomic DNA sequences will typically contain both coding and noncoding regions, which cannot be analyzed by the traditional dynamical programming algorithm.[1] As a protein evolves slower than its coding DNA, it will be more reliable to align the protein than the underlying DNA, and an algorithm that compares genomic DNA should incorporate the information from the protein. Presently, this problem is solved by separating the sequences into coding and noncoding parts, then analyzing them separately, and, finally, patching the resulting alignments into a global alignment. This is laborious and cannot be done if the DNA has overlapping reading frames. In this chapter we present an algorithm that solves these problems.

It should be noted that all evolutionary events happen at the DNA level, as proteins do not replicate. The basic events are (1) substitutions, which in coding regions can have the additional consequence of changing

[1] S. B. Needlemann and C. D. Wunsch, *J. Mol. Biol.* **48**, 444 (1970).